Wireless Enhancements for Storage Area Networks

David Griffith, Kotikalapudi Sriram, JingSi Gao, and Nada Golmie

***Abstract*— We propose the creation of a wireless storage area network (SAN) and analyze its benefits. The proposed wireless SAN (WSAN) consists of a SAN switch that is connected to multiple wireless access points (APs) that communicate with the storage devices. This network would save space and reduce overall costs by not requiring wired connections. Wireless SANS would also provide more freedom in the placement of storage devices. However, because the number of wireless access points is less than the number of storage devices, it is possible for user data requests to be blocked if all access points are busy. An important design goal is therefore to minimize the probability that a network access request will be blocked.**

***Index Terms*— storage area networks, wireless access points, network simulation, performance metrics**

I. Introduction

A storage area network (SAN) is a special-purpose network that interconnects data storage devices with associated data servers on behalf of a larger network of users [1], [2]. Storage devices are connected to a central SAN switch that in turn is connected to an outside network that allows remote users to access the storage devices. Storage networks are evolving toward using IP-based solutions such as Internet to Small Computer System Interface (iSCSI), Fibre Channel over IP (FCIP), and Internet Fibre Channel Protocol (iFCP) [3]. An example SAN configuration is shown in Fig. 1. Users access the network using applications such as FTP and can either upload data to the storage devices or retrieve data for another application's use. Larger SANs allow users to connect to the storage devices through a network of interconnected SAN switches. The disk arrays and tape silos that compose the SAN are typically housed in a rented facility and must be widely spaced to allow access for maintenance and upgrades. This can lead to considerable capital costs in terms of floor space and the amount of cabling required.

In order to reduce the network complexity, we wish to decrease the amount of cabling between the switch and the storage devices. One way to do this, which we explore in this paper, is to allow the storage disk controllers to connect to the SAN switch over wireless channels. Ours is the first effort of which we are aware that examines using wireless links to connect SAN switches to storage devices. Other work has considered optical links between SAN switches to form ad hoc networks [4], using IEEE 802.11 links to transfer video from digital cameras to SANs [5], and mechanisms to allow wireless devices to efficiently store and retrieve data on external storage devices [6]. Wireless links were used in Canada to allow a credit union to connect its main data center to a remote (wired) SAN that functions as a disaster recovery site [7].

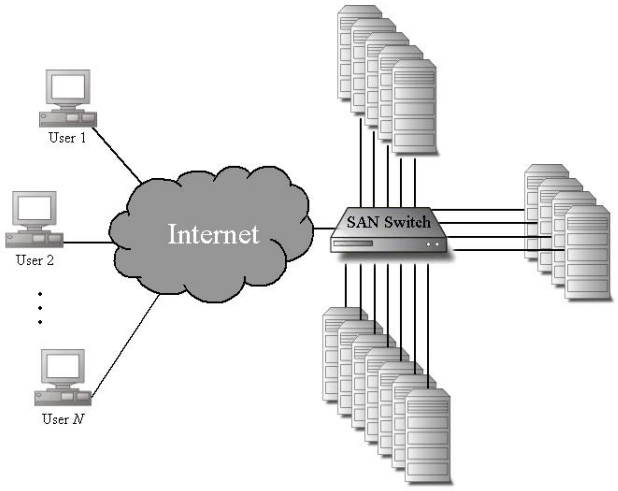

Fig. 1. Example of a storage area network with direct connections from the SAN switch to each storage device.

In Fig. 2, we show how the example SAN from Fig. 1 can be modified by incorporating wireless access points (APs). This can reduce operational costs by eliminating cabling and allowing the possibility of tighter equipment packing. Because introducing wireless access to the SAN reduces the number of connections between the storage devices and the central switch, the network must be designed to prevent a degradation in the quality of service provided to the users.

To model the wireless SAN (WSAN) performance, we developed a theoretical model of the network characteristics using queuing theory. We next created a C++ simulation of a wireless SAN and compared the results that we obtained from the simulation with results obtained from the theoretical model in order to verify the accuracy of our analysis. The simulation was used to study how the number of available access points and various traffic loads affects the overall blocking probability and compare the wireless SAN's cost with the cost of traditional non-wireless SANs. It was found that wireless SANs would be much more cost effective. The results of the simulation can be used to provide design guidelines for building future wireless SANs.

The rest of this paper is organized as follows. In Section II, we describe the queuing model that we developed to determine

D. Griffith, K. Sriram, and N. Golmie are with the National Institute of Standards and Technology (NIST), Gaithersburg, MD 20899 USA (contact e-mail: david.griffith@nist.gov).

J.S. Gao is with the Department of Electrical & Computer Engineering, University of Maryland, College Park, MD 20742 USA

This research was partially supported by the National Communications System (NCS) and the National Science Foundation (NSF).

Icons used in Fig. 1 and Fig. 2 are courtesy of The Storage Networking Industry Association (SNIA) located at the following URL: http://www.snia.org/about/images/network_component_icons/.

The identification of any commercial firm, product, or trade name in this paper does not imply endorsement or recommendation by the National Institute of Standards and Technology.

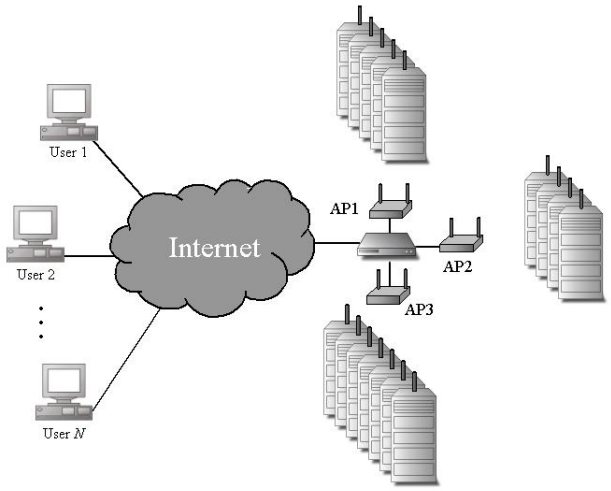

Fig. 2. Example of a storage area network in which each storage device is connected to the SAN switch via a wireless access point.

the effect of the number of access points and the network load on the probability that a request for service will be blocked. In Section III, we describe the methodology that we used in constructing a set of simulations in C++ that we used to verify the accuracy of the theoretical analysis and to determine the optimal number of access points for giving network load and a maximum desired service request blocking probability. In Section IV, we discuss the simulation results and examine how burstiness in the service request arrival process can lead to a degradation in performance unless an adequate number of access points are provided. We also use a simple cost model to demonstrate that using wireless access points can lead to significant savings over the lifetime of the network. We summarize the major findings of this paper in Section V.

II. Analysis

We assume that requests for service arrive at the SAN switch according to a Poisson process with rate λ. The size of the file to be uploaded or downloaded in connection with the ith service request is S_i blocks, where the size of a single block is 2108 bytes. Fig. 3 shows the possible outcomes for an arriving service request. The SAN switch has an onboard memory cache that can be used to store frequently accessed files. P_1 is the probability that the service request can be serviced by using the cache. The cache service time for the ith service request is $\tau_1(i) = S_i\varphi_1$, where φ_1 is the time required for the SAN switch cache to process a single 2108 byte block of data, measured in seconds. Because service requests are diverted to the switch cache with probability P_1, as shown in Fig. 3, the remaining requests that reach the access points can be characterized by a Poisson process with rate $\lambda' = (1-P_1)\lambda$.

If the service request cannot be handled by the switch cache, and if at least one access point is free, the request will be directed to one of the storage devices. We assume that each storage device can be reached via any of the access points. At the storage device, files can be stored either in the memory cache or on the disk drive, depending on the

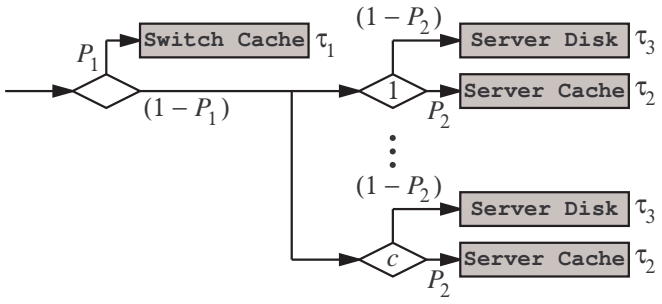

Fig. 3. Service request flow for wireless SAN.

frequency with which they are accessed. The memory cache allows frequently requested files to be accessed in less time than would be required if they were stored on the disk drive, but it does not have as much storage capacity. We define P_2 to be the probability that the request can be serviced by using the onboard memory cache in the selected storage device rather than by using the hard drive, as shown in Fig. 3.

The data located on the storage device is delivered to the user over a wireless link. If the link bandwidth B, measured in blocks/sec, is sufficiently low, it will determine the amount of time required to complete the service request, rather than the cache or disk read times at the storage device. Conversely, if the wireless link bandwidth is very high then the amount of time required to deliver the requested data will be determined by the disk or cache read rate. The cache service completion time for the ith service request is therefore $\tau_2(i) = \max(S_i\varphi_2, S_i/B)$, where φ_2 is the amount of time that the cache requires to read one block of data. Similarly, the service completion time associated with the disk drive is $\tau_3(i) = \max(S_i\varphi_3, S_i/B)$, where φ_3 is the amount of time that the disk requires to read one block of data. The average service request completion rate at each access point, μ, is the inverse of the mean service request completion time at a storage device, τ_{avg}. Using Fig. 3, we see that

$$\tau_{\text{avg}} = (1-P_1)[P_2\tau_2 + (1-P_2)\tau_3], \tag{1}$$

where $\tau_2 = \max(\mu_S\varphi_2, \mu_S/B)$ and $\tau_3 = \max(\mu_S\varphi_3, \mu_S/B)$ are the average cache and disk service times and μ_S is the average requested file size. In the simulations we assumed that the access points used the IEEE 802.11a standard, which supports a maximum data rate of 35 Mb/s. We modeled the effect of a large number of access points on the available bandwidth by defining the channel bandwidth to be 35 Mb/s if the number of access points is eight or fewer.

We define the WSAN utilization to be $\rho = \lambda/c\mu$, where λ is the arrival rate and $\mu = \tau_{\text{avg}}^{-1}$ is the service rate. Because the access points can be modeled as an $M/G/c/c$ queuing system that is accessed with probability $1 - P_1$, the service request blocking probability, which is the probability that all access points are busy, is given by the modified Erlang-B loss formula [8]

$$P_B = B(c,\rho) = \frac{(1-P_1)(c\rho)^c/c!}{\sum_{n=0}^{c}(c\rho)^n/n!}. \tag{2}$$

III. Simulation Model

We use a simulation model, implemented in C++, that incorporates discrete event processing. The simulation keeps track of the number of access points that are in use, the time when each active access point will become free, and the time of arrival of the next service request. We set the number of servers in the network to be 30. At the beginning of each iteration, the simulation checks to see if the service request arrival time is greater than the completion times of any of the wireless access points. Any access points that meet this criterion are freed for possible use by the service request. Next, the simulation implements the decision algorithm depicted in the flowchart in Fig. 5. The simulation first generates a random number and compares it to a threshold, P_1, to determine whether the request can be satisfied by using the cache in the SAN switch. If the random number is less than P_1, the request is serviced by the switch cache. We set $P_1 = 0.2133$ for all the simulation runs. The cache can read out data at a rate of $\varphi_1 = 120$ μsec per 2108 byte block.

If a storage device is required, the simulation determines whether an access point is available. If no access point is available, the request is considered blocked and the simulation increments a tally variable. If an access point is available, the simulation generates a random number to determine the size of the requested file. Fig. 4 shows the probability mass function of the requested file size; the average file size is 44.9 Mbytes. Once the file size is known, the simulation generates another

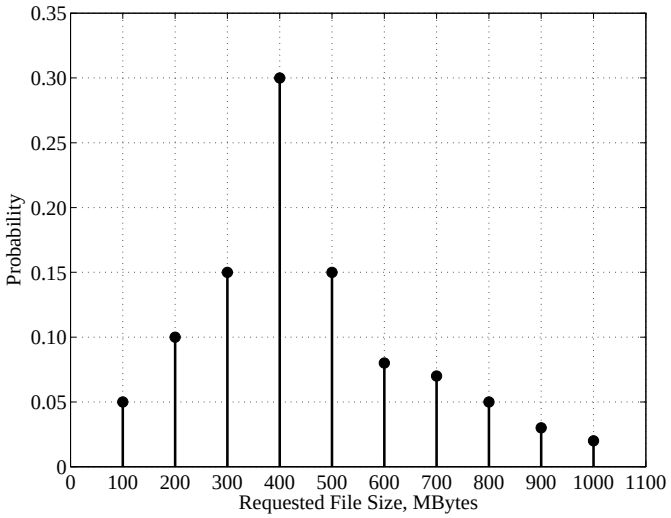

Fig. 4. Probability mass function for service completion times at each SAN storage device.

random number and compares it to the threshold $P_2 = 0.64$ to determine whether the request will be serviced by the disk (if the random number is greater than P_2) or the cache in the storage device (if it is less than P_2). The amount of time required to complete the request is computed based on the outcome, using $\varphi_2 = 180$ μsec and $\varphi_3 = 10000$ μsec as the read times for one block of data for the cache and disk, respectively. Using equation (1), we can compute the average service request completion time at a storage device, given this set of constants, to be $\tau_{\text{avg}} = 16.1$ msec. The simulation adds the computed service time for a given request to the current simulation time to determine the time when the reserved access point will become free. Once the service time for the request is computed, provided that the request is not been blocked, the simulation adds the computed service time to the total service time for the WSAN and begins the next iteration by computing the next service request arrival time. It continues this process until a desired number of service requests have been processed.

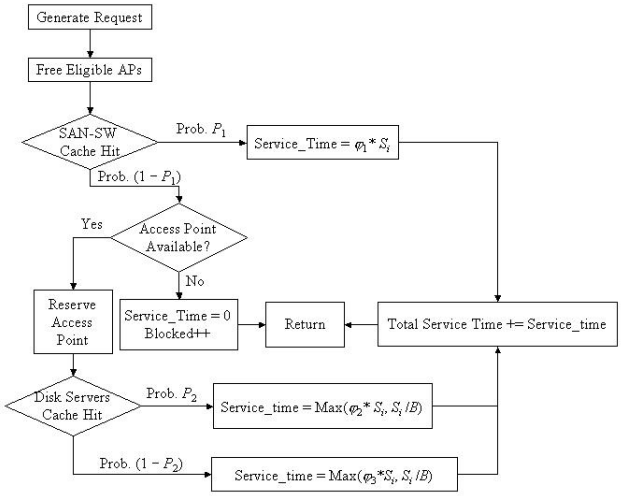

Fig. 5. Flowchart for processing arriving service requests.

The connection arrival process is assumed to be either a Poisson process (smooth arrivals) or a bursty process represented by hyper-exponential interarrival times. The bursty nature of I/O requests in storage networks is well-established. In [9], it was shown that request arrivals do not follow a Poisson process. The self similar nature of request arrivals was demonstrated in [10]. Examples of bursty request arrivals observed in real-world storage systems may be found in [11].

The hyper-exponential density allows each inter-arrival time to be distributed according to one of M exponential random variables, independently of all the other inter-arrival times in the arrival process. For $M = 2$ arrival modes, the hyper-exponential density function is

$$f_X(x) = p\lambda_1 \exp(-\lambda_1 x) + (1-p)\lambda_2 \exp(-\lambda_2 x), \qquad (3)$$

where we assume without loss of generality that $\lambda_1 > \lambda_2$, and where p is the probability of being in mode 1 where arrivals occur at the higher rate of λ_1, and $(1-p)$ is the probability of being in mode 2 where arrivals occur at the lower rate of λ_2. Two parameters that are used to characterize the hyper-exponential arrival process are the ratio of arrival rates, θ, and the average arrival rate, λ. These parameters are

$$\theta = \lambda_1/\lambda_2 \qquad (4)$$

and

$$\lambda = \left(p\lambda_1^{-1} + (1-p)\lambda_2^{-1}\right)^{-1}, \qquad (5)$$

respectively. A parameter that is used to measure the amount of burstiness in the arrival process is the coefficient of variation,

which is

$$\chi^2 = \frac{\mathsf{Var}\{X\}}{(\mathsf{E}\{X\})^2} = \frac{2(p+(1-p)\theta^2)}{(p+(1-p)\theta)^2} - 1. \quad (6)$$

If $p = 0$ or $p = 1$, then $\chi^2 = 1$, indicating that we have an exponential distribution with mean $1/\lambda_2$ or $1/\lambda_1$, respectively.

We chose the parameters of the hyper-exponential distribution to obtain different values of the service request arrival rate λ and burstiness χ^2 in the simulations. In addition to modeling exponential inter-arrival times, we modeled a moderately bursty arrival process by choosing p = 0.9, θ = 10 to give χ^2 = 5.04. We also modeled a very bursty arrival process by choosing p = 0.95 and θ = 40, which give χ^2 = 17.6.

In Fig. 6, we plot samples of hyper-exponential processes with an average arrival rate of 1 arrival per second over a period of 200 seconds. In Fig. 6(a), we show a Poisson process, whose coefficient of variation is unity. While there is some clustering of arrivals, they tend to be spread uniformly in time. In contrast, the hyper-exponential process shown in Fig. 6(b), for which $\chi^2 = 5.04$, features more clustering of arrivals and larger gaps between clusters. In Fig. 6(c), we see even greater clustering taking place and very large gaps between the groups of densely packed arrivals; the coefficient of variation is 17.6 in this case.

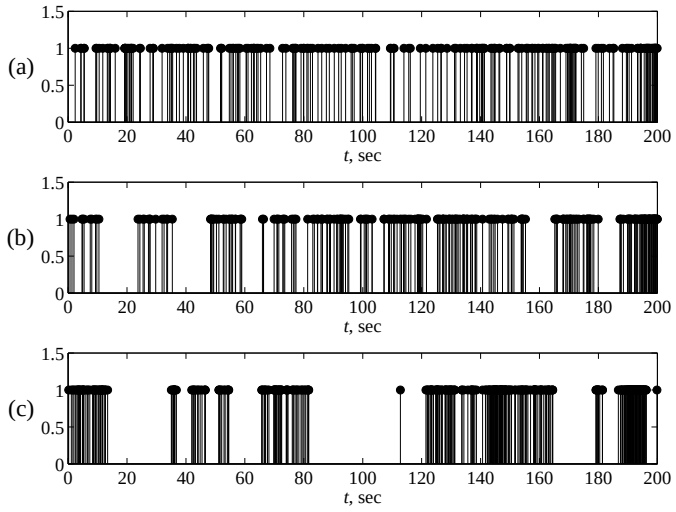

Fig. 6. Sample hyper-exponential processes with average arrival rate λ = 1 arrival/sec and coefficient of variation (a) $\chi^2 = 1.0$, (b) $\chi^2 = 5.04$, (c) $\chi^2 = 17.6$.

IV. Results and Discussion

We first verify the performance of our simulation tool by comparing the results of the $M/G/c/c$ analysis in Section II with results obtained from the simulations described in Section III. Table I and Table II show the service request blocking probability versus the utilization, ρ, using six and eight access points, respectively. For the six access point scenario, the percent deviation of the experimental results from the corresponding theoretical value was on the order of 0.1% to 1%. We obtained similar results for the eight access point scenario. The worst-case error was 10%, which occurred in only one of the cases we examined (eight access points, $\rho = 0.3$). In several cases, the measured error is recorded in the tables as zero because the deviation of the experimental results from the theoretical prediction was less than four significant figures. These results demonstrate the accuracy of the simulation tool that we will use to produce the rest of the results in this section.

TABLE I
Theoretical and experimental values of blocking probability for 6 access points.

ρ	Experiment	$B(c,\rho)$	% Error
0.20	1.000E-03	1.000E-03	0.000E+00
0.25	3.000E-03	2.800E-03	7.143E+00
0.30	6.000E-03	6.200E-03	3.226E+00
0.35	1.140E-02	1.150E-02	8.696E-01
0.40	1.880E-02	1.920E-02	2.083E+00
0.45	2.890E-02	2.900E-02	3.448E-01
0.50	4.130E-02	4.100E-02	7.317E-01
0.55	5.500E-02	5.480E-02	3.650E-01
0.60	6.960E-02	7.010E-02	7.133E-01
0.65	8.710E-02	8.650E-02	6.936E-01
0.70	1.036E-01	1.037E-01	9.643E-02
0.75	1.200E-01	1.213E-01	1.072E+00
0.80	1.403E-01	1.391E-01	8.627E-01
0.85	1.570E-01	1.568E-01	1.276E-01
0.90	1.751E-01	1.744E-01	4.014E-01
0.95	1.918E-01	1.916E-01	1.044E-01

TABLE II
Theoretical and experimental values of blocking probability for 8 access points.

ρ	Experiment	$B(c,\rho)$	% Error
0.20	2.000E-04	2.000E-04	0.000E+00
0.25	7.000E-04	7.000E-04	0.000E+00
0.30	1.800E-03	2.000E-03	1.000E+01
0.35	4.300E-03	4.500E-03	4.444E+00
0.40	8.800E-03	8.800E-03	0.000E+00
0.45	1.520E-02	1.520E-02	0.000E+00
0.50	2.390E-02	2.390E-02	0.000E+00
0.55	3.470E-02	3.490E-02	5.731E-01
0.60	4.850E-02	4.790E-02	1.253E+00
0.65	6.380E-02	6.270E-02	1.754E+00
0.70	7.800E-02	7.880E-02	1.015E+00
0.75	9.590E-02	9.590E-02	0.000E+00
0.80	1.134E-01	1.136E-01	1.761E-01
0.85	1.321E-01	1.316E-01	3.799E-01
0.90	1.496E-01	1.497E-01	6.680E-02
0.95	1.680E-01	1.677E-01	1.789E-01

The tables also indicate the maximum utilization that can be supported by a given number of access points, assuming that service requests conform to a Poisson model. For example, if the SAN switch is connected to 6 access points and the maximum acceptable request blocking probability is 0.001, then the access point utilization cannot exceed 0.2. Given that the mean service time is $\tau_{\text{avg}} = 16.1$ msec, this means that the mean request arrival rate can be at most $\lambda = 83.8$ requests/hour. Likewise, the maximum average request arrival rate is $\lambda = 139.7$ requests/hour if eight access points are used and the maximum acceptable request blocking probability is 0.001, since the maximum allowable utilization in this case is 0.25.

We next consider the effect of burstiness in the request arrival process. In Fig. 7 we plot the service request blocking probability as a function of the number of access points used by a WSAN for four values of λ, the request arrival rate. We used a hyper-exponential arrival process with $\chi^2 = 5.04$. We see from the figure that using 4 access points is not sufficient to support a blocking probability of less than 0.001 for any of the arrival rates that we considered. Using the value of μ that we previously computed, we find that an arrival rate of 20 requests/hour gives $\lambda' = 15.734$ requests/hour, which for $c = 4$ corresponds to a utilization of $\rho = \lambda'/c\mu = 0.0716$. Note that, for the same maximum acceptable blocking probability of 0.001, the WSAN supports a lower mean request arrival rate than in the Poisson case. For example, assuming a maximum acceptable blocking probability of 0.001, the maximum supportable average arrival rate is approximately 40 requests/hour if six access points are being used and it is approximately 50 requests/hour if eight access points are available. This represents a significant decrease in the amount of traffic that the WSAN can handle with respect to the Poisson arrival scenario, and indicates that accurate characterization of the arrival process is crucial to the WSAN design process.

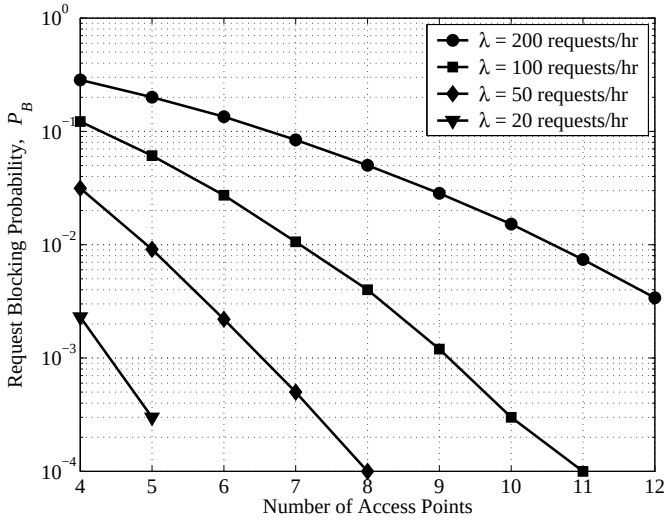

Fig. 7. Blocking probability versus number of access points for different request arrival rates using Poisson arrivals.

Fig. 8 shows the effect of burstiness in the request arrival process on the WSAN's blocking probability, P_B, which is plotted vs. the number of access points for three values of χ^2. The utilization in all three cases was 0.5. Because of the high utilization, the blocking probability curves do not exhibit the steep rolloff that we observed in connection with lower utilization levels in Fig. 7. Fig. 8 shows that an increase in χ^2 produces a corresponding increase in the blocking probability. The performance degradation, measured as the increase in blocking probability, associated with increasing χ^2 from 5.04 to 17.6 varies from 1.9 times greater (when the number of access points is 4) to 3.2 times greater (when the number of access points as 12) than the degradation associated with increasing χ^2 from 1 to 5.04. This is reflected in the decreasing magnitude of the slope of the blocking probability curve as χ^2 increases.

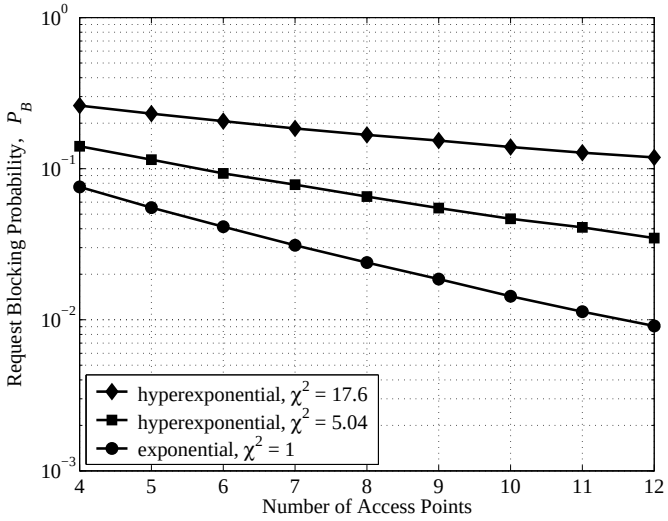

Fig. 8. Blocking probability versus number of access points using hyper-exponential arrival processes with varying degrees of burstiness.

To further illustrate the effect of burstiness on the ability of the WSAN to process service requests, we graph the blocking probability vs. the utilization, ρ, in Fig. 9. We consider the following two scenarios: a WSAN using six access points and one using eight access points. We use a Poisson arrival process and a hyper-exponential arrival process with coefficient of variation $\chi^2 = 5.04$ for each of the two scenarios, giving us the four curves shown in the figure. From the figure, we see that the introduction of burstiness causes a noticeable performance degradation, so that the request blocking probability that we obtained using eight access points with a hyper-exponential arrival process was greater than the blocking probability using six access points and a Poisson arrival process.

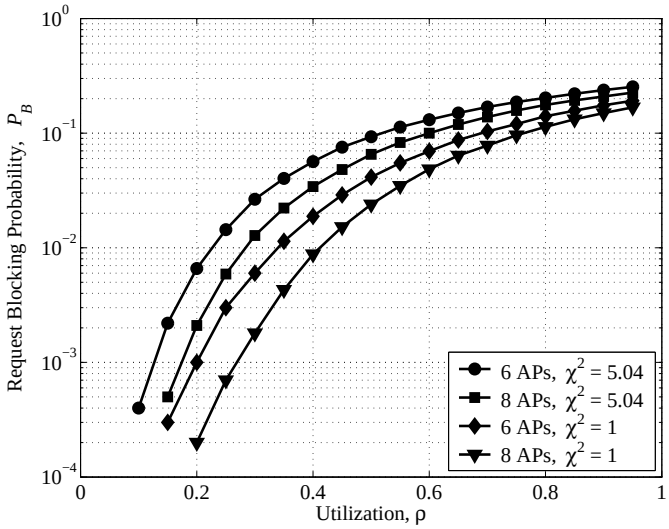

Fig. 9. Blocking probability vs. utilization for hyper-exponential arrival processes with $\chi^2 = 1.0$ and $\chi^2 = 5.04$.

Fig. 10 also illustrates the impact of bursty arrivals on WSAN performance. In the figure, we plot the service request

blocking probability versus the number of access points at five different levels of utilization. We use a hyper-exponential arrival process with $\chi^2 = 5.04$. And moderately high utilization ($\rho = 0.7$) the blocking probability remains above 0.1, even when 12 access points are being used. Decreasing the utilization does not produce a significant improvement in performance. When $\rho = 0.3$, at least nine access points are required in order to have a blocking probability of less than 0.01. When 12 access points are in use, the blocking probability still remains above 0.001. These results, along with the results in the preceding figures, show that even moderate burstiness can produce performance degradations that require the use of additional access points or scaling the network so that the utilization is kept within certain limits.

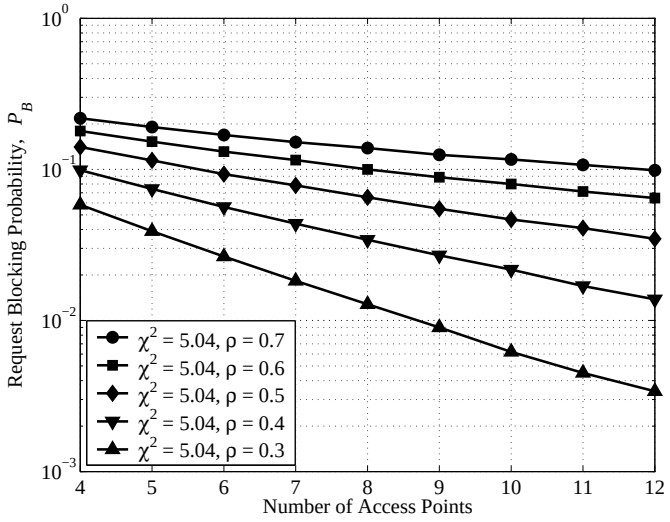

Fig. 10. Blocking probability versus number of access points with hyper-exponential arrival processes ($\chi^2 = 5.04$) with varying utilization.

Finally, we examine the cost savings that are possible if a wireless network is used to connect the SAN switch to the storage devices rather than cabling. The savings result from lower equipment costs associated with less cabling and a smaller SAN switch that needs to accommodate on the order of 10 ports rather than dozens. Performance analysis of different inter-floor SAN switch network topologies in [12] has demonstrated that the best performance can be achieved by using wraparound links to connect the switches on the top floor to those on the bottom floor. In addition, a star topology consisting of center switches and arm switches should be used on each floor. Our cost comparison is independent of the switch interconnection scheme.

For our cost computations, we assume that the base price of renting a single floor to house the SAN is \$2000 and the base price of a SAN storage device is \$100. The cost of the SAN switch, cabling, and access points is modeled by assigning a per-port cost of \$50 to a wired switch and a per-port cost of \$70 to a switch that is connected to wireless access points. Thus, if F is the number of floors in the facility that is hosting the SAN, K is the number of storage devices per floor, and N is the number of wireless access points per floor, the costs associated with a wired SAN and WSAN are respectively

$$C_{\mathrm{SAN}}(K,F) = (2000 + 150K)F \quad (7)$$

$$C_{\mathrm{WSAN}}(N,K,F) = (2000 + 100K + 70N)F \quad (8)$$

In Fig. 11 we plot equations (7) and (8) for the case where $K = 30$, $N = 8$, and the number of floors, F, varies from 1 to 5. These results show that the cost associated with deploying a WSAN are lower than the costs associated with deploying a SAN of the same capacity, and that the WSAN's advantage increases as the size of the network increases.

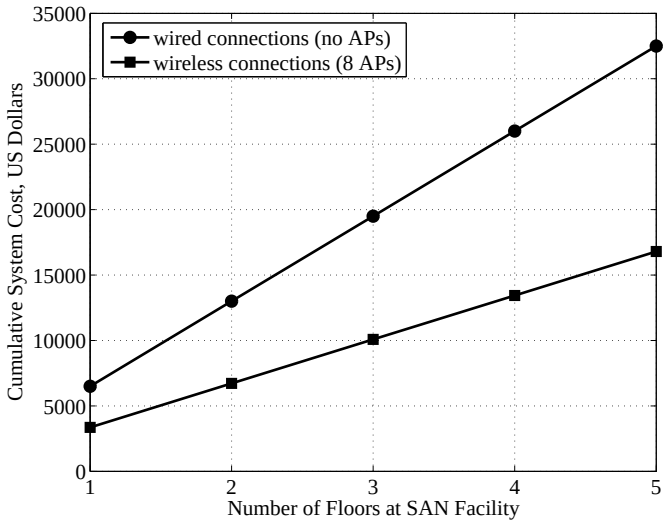

Fig. 11. Cost of SAN and WSAN deployments over a five-year period.

In general, if the cost per port for a SAN is S_{SAN} and the cost per port for a WSAN is S_{WSAN}, then from equations (7) and (8) we find that the WSAN will be less expensive if

$$\frac{N}{K} < \frac{S_{\mathrm{SAN}}}{S_{\mathrm{WSAN}}} = \zeta. \quad (9)$$

In Fig. 11, $\zeta = 0.7143$. For this case, where $K = 30$, as long as $N < 22$, the WSAN will cost less than the SAN. We have seen from the other results in this section that that number is sufficient to achieve a request blocking probability that is well below 0.001, especially if the utilization is relatively low and if the arrival process is relatively smooth.

V. Summary

In this paper, we examined the effect of replacing the circuit switched architecture for storage area networks with one based on wireless access points. We developed a theoretical model that allows us to compute the probability that a request for service will be blocked given a certain level of demand and a known the number of access points. We used this model to show showed that there are significant quantitative performance and cost benefits that result from the incorporation of a shared wireless medium between the central switch and the storage devices. Even greater benefits can be achieved when the power consumption and form factor considerations are incorporated into the cost models. These models can be extended to examine the different kinds of

wireless technologies that are currently available (e.g., IEEE 802.11b, GPRS, UMTS, IEEE 802.16d).

REFERENCES

[1] Milanovic, S. and Petrovic, Z., "Building the enterprise-wide storage area network," *Proceedings of the International Conference on Trends in Communications (EUROCON 2001)*, vol. 1, pp. 136-139, 4-7 July 2001.

[2] DeCusatis, C., "Storage area network applications," *Proceedings of the Optical Fiber Communication Conference and Exhibit (OFC 2002)*, pp. 443-444, 17-22 March 2002.

[3] Knowles, M., "Survey of the storage evolution," *Proceedings of User Group Conference, 2003*, pp. 362-367, 9-13 June 2003.

[4] Hui, J.Y., "Wireless optical ad-hoc networks for embedded systems," *Proceedings of the IEEE International Conference on Performance, Computing, and Communications, 2001*, pp. 140-144, 4-6 April 2001.

[5] Corcoran, P.M. and Raducan, I., "Practical aspects of video data transfer to network storage using 802.11 WLAN technology for low-end consumer digital cameras," *IEEE Transactions on Consumer Electronics*, Vol. 49, No. 4, pp. 902-910, Nov. 2003.

[6] Srinivasan, S.H., "A Thin Storage Architecture for Wireless," *Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshops, 2005 (PerCom 2005 Workshops)*, pp. 163-167, 8-12 March 2005.

[7] Mearian, L., "Canadian credit union installs wireless SAN link," Computerworld, 24 March 2003.

[8] Gross, D., and Harris, C.M., *Fundamentals of Queuing Theory*, 2nd Ed., John Wiley & Sons, 1985.

[9] Ganger, G., "Generating representative synthetic workloads," *Proceedings of the Computer Measurement Group Conference*, pp. 1263-1269, December 1995.

[10] Gómez, M. and Santonja, V., "Self-similarity in IIO workload: analysis and modeling," *Workload Characterization: Methodology and Case Studies. IEEE Computer Society*, pages 97-104, 1998.

[11] Molero, X., Silla, F., Santonja, V., and Duato, J., "Modeling and simulation of storage area networks," *Proceedings of the 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2000*, pp. 307-314, 29 Aug.-1 Sept. 2000.

[12] Molero, X., Silla, F., Santonja, V., and Duato, J., "On the scalability of topologies for storage area networks in building environments," *Proceedings of the IEEE International Symposium on Network Computing and Applications, 2001 (NCA 2001)*, pp. 332-335, 8-10 Oct. 2001.